

Отчет по практической работе №1

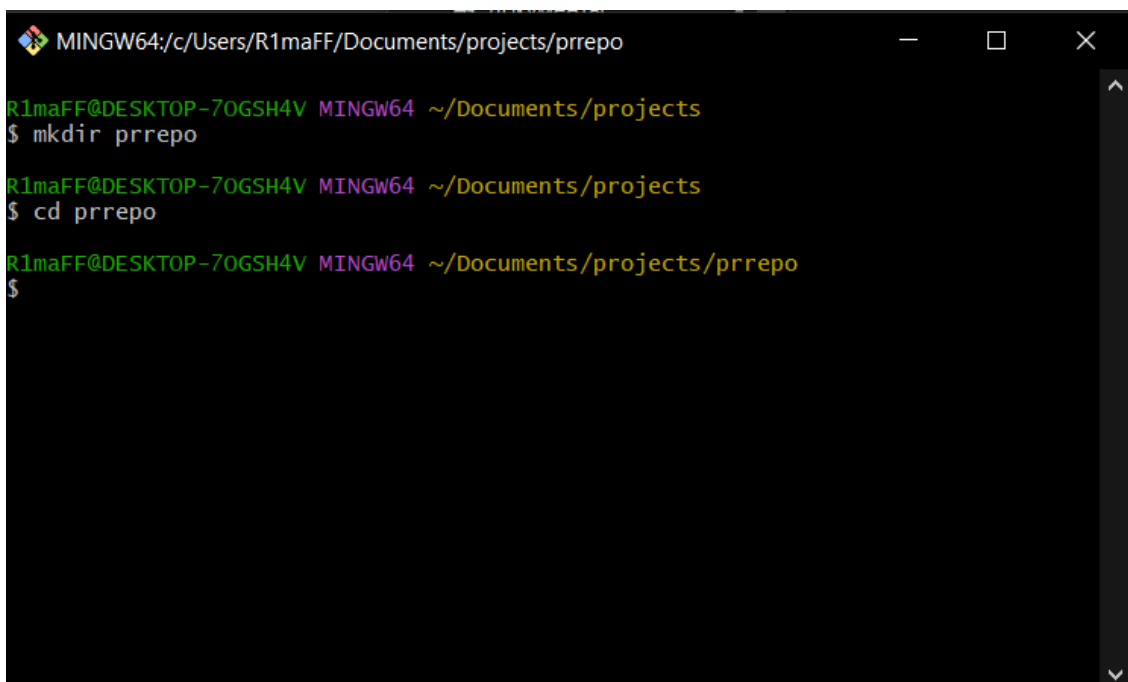
Основы Git и Github

Цель работы:

1. изучение наиболее распространенных практик в области контроля версий программного обеспечения, его использования в командной разработке ПО и DevOps;
2. изучение концепции Git, основанной на понятиях репозитория и ветвления версий ПО;
3. изучение порядка использования GitHub и его базовых операций.

Ход работы:

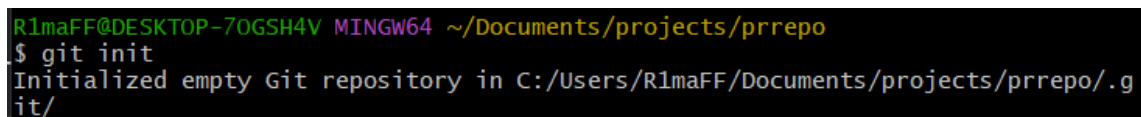
Создаем новый каталог prrepo и переходим в него.



```
MINGW64:/c/Users/R1maFF/Documents/projects/prrepo
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects
$ mkdir prrepo
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects
$ cd prrepo
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo
$
```

Рис 1.1

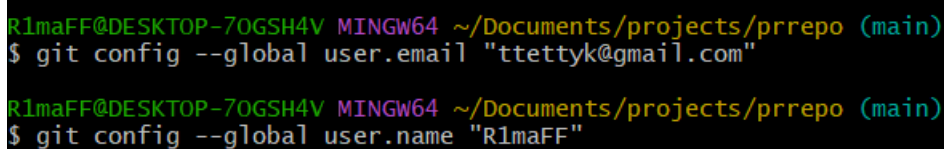
Создаем локальный репозиторий.



```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo
$ git init
Initialized empty Git repository in C:/Users/R1maFF/Documents/projects/prrepo/.git/
```

Рис 1.2

Перед созданием README файла сообщаем git наши данные.



```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ git config --global user.email "ttettyk@gmail.com"
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ git config --global user.name "R1maFF"
```

Рис 1.3

Создаем README файл и добавляем его в ветку main.

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ touch README.md

R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ git add README.md
```

Рис 1.4

Фиксируем изменения.

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ git commit -m "Initial commit"
[main (root-commit) 265c7bf] Initial commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
```

Рис 1.5

Создаем новую ветку newbranch и делаем ее активной.

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ git branch newbranch

R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ git checkout newbranch
Switched to branch 'newbranch'
```

Рис 1.6

Создаем newbranchfile и добавляем его в ветку newbranch

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (newbranch)
$ touch newbranchfile

R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (newbranch)
$ git add newbranchfile
```

Рис 1.7

Фиксируем изменения и сразу после этого отменяем последний commit.

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (newbranch)
$ git commit -m "Добавлен файл newbranchfile"
[newbranch 1a32cf0] Добавлен файл newbranchfile
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newbranchfile

R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (newbranch)
$ git reset HEAD~1
```

Рис 1.8

Создаем новый файл newgoodfile, добавляем его в ветку и фиксируем изменения.

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (newbranch)
$ touch newgoodfile

R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (newbranch)
$ git add newgoodfile

R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (newbranch)
$ git commit -m "Добавлен файл newgoodfile"
[newbranch 5d805af] Добавлен файл newgoodfile
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newgoodfile
```

Рис 1.9

Делаем активной ветку main и объединяем изменения в новой ветке с основной.

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (newbranch)
$ git checkout main
Switched to branch 'main'

R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ git merge newbranch
Updating 265c7bf..5d805af
Fast-forward
 newgoodfile | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newgoodfile
```

Рис 1.10

Репозиторий другого студента.

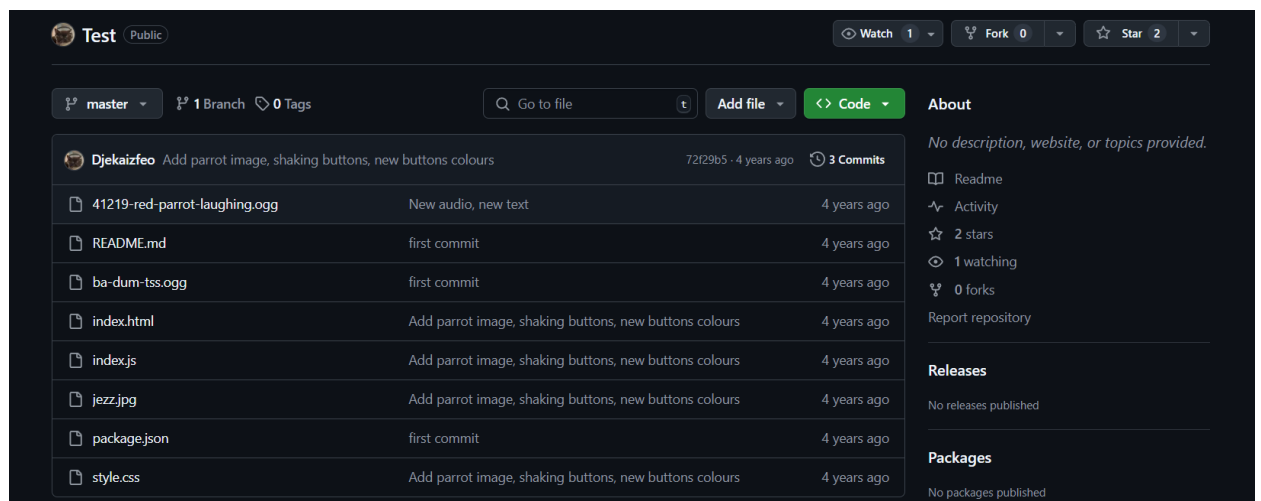


Рис 1.11

Fork репозитория другого студента.

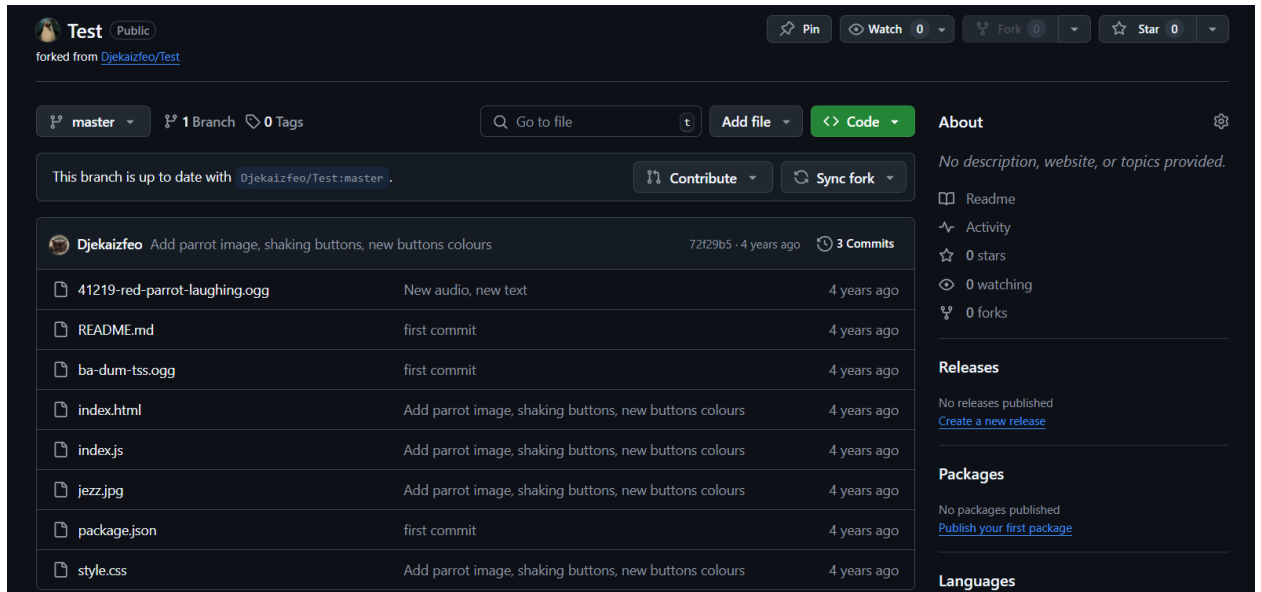


Рис 1.12

Clone проекта в локальный репозиторий.

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ git clone https://github.com/R1maFF/Test
Cloning into 'Test'...
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 19 (delta 7), reused 15 (delta 3), pack-reused 0 (from 0)
Receiving objects: 100% (19/19), 349.96 KiB | 756.00 KiB/s, done.
Resolving deltas: 100% (7/7), done.
```

Рис 1.13

Переход в папку клонированного проекта, создание в нем нового файла и фиксирование изменений.

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo (main)
$ cd Test
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo/Test (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo/Test (master)
$ touch newfile.txt
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo/Test (master)
$ git add newfile.txt
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo/Test (master)
$ git commit -m "Добавлен файл newfile.txt"
[master 1b42b9b] Добавлен файл newfile.txt
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newfile.txt
```

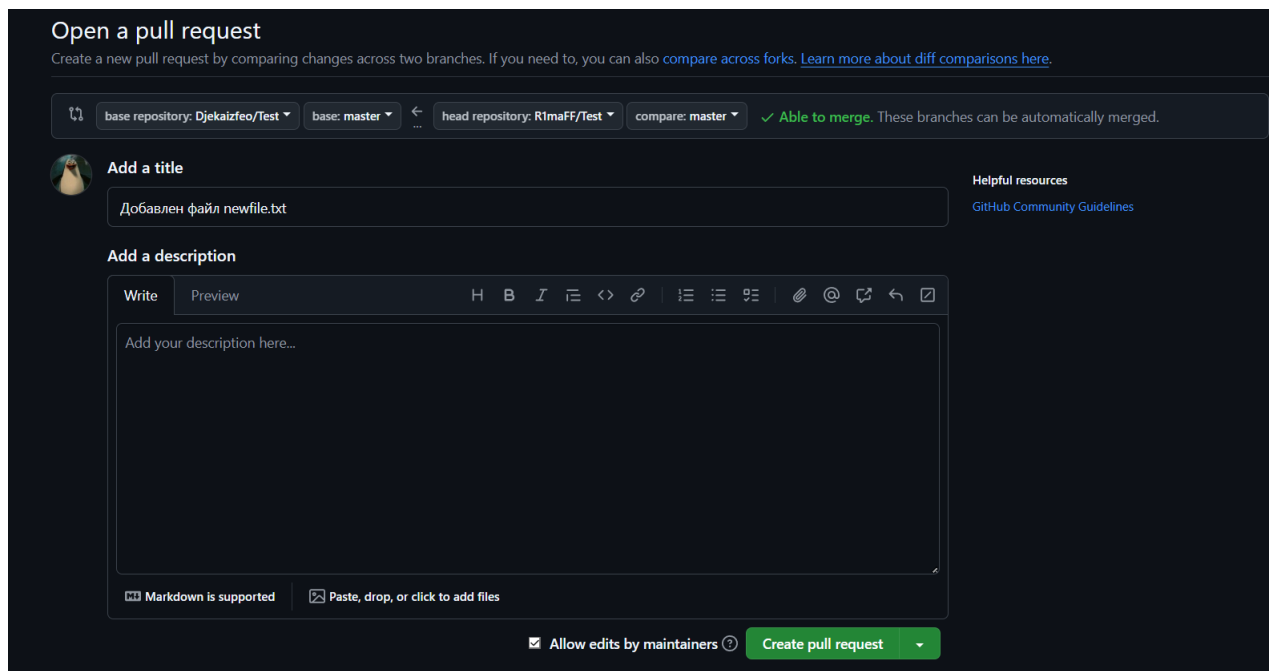
Рис 1.14

Синхронизация с fork-репозиторием.

```
R1maFF@DESKTOP-70GSH4V MINGW64 ~/Documents/projects/prrepo/Test (master)
$ git push origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 304 bytes | 304.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/R1maFF/Test
72f29b5..1b42b9b master -> master
```

Рис 1.15

Формирование pull request к origin проекту на прием изменения.



Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#). [Learn more about diff comparisons here](#).

base repository: Djekaizfo/Test base: master head repository: R1maFF/Test compare: master ✓ Able to merge. These branches can be automatically merged.

Add a title

Добавлен файл newfile.txt

Add a description

Write Preview

Add your description here...

Markdown is supported Paste, drop, or click to add files

Allow edits by maintainers Create pull request

Рис 1.16

Вывод: В результате практической были изучены базовые операции Github и порядок его использования. Был создан репозиторий с несколькими ветками и файлами. Выполнен процесс клонирования проекта другого студента, добавления в него файла и отправки pull request к origin проекту.